

a.)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-312126

(43) 公開日 平成11年(1999)11月9日

(51) Int.Cl.⁶

G 0 6 F 13/00
12/08

識別記号

3 0 1

3 2 0

F I

G 0 6 F 13/00
12/08

3 0 1 P
J
G
3 2 0

審査請求 未請求 請求項の数 8 F D (全 23 頁)

(21) 出願番号 特願平10-132712

(22) 出願日 平成10年(1998)4月27日

(71) 出願人 000005108

株式会社日立製作所
東京都千代田区神田駿河台四丁目6番地

(72) 発明者 山神 憲司

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

(72) 発明者 藤本 和久

東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内

(72) 発明者 黒須 康雄

神奈川県小田原市国府津2880番地 株式会社日立製作所ストレージシステム事業部内

(74) 代理人 弁理士 笹岡 茂 (外1名)

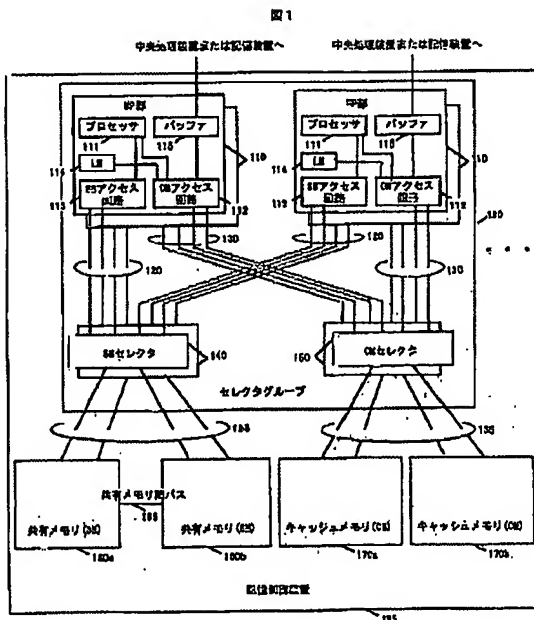
最終頁に続く

(54) 【発明の名称】 記憶制御装置

(57) 【要約】

【課題】 複数プロセッサと外部メモリを接続する場合のL S Iピンネックおよびパッケージのコネクタネックの解消を図る。

【解決手段】 中央処理装置と記憶装置間でデータの入出力を制御する記憶制御装置195は複数のセレクトグループ190と、共有メモリ間バス165で接続された共有メモリ160a, 160bと、キャッシュメモリ170a, 170bからなり、上記各190は、図の例では、4つのMP部110、2つのS Mセクタ140、2つのC Mセクタ150を有し、各MP部110はプロセッサ、L M(ローカルメモリ)114、S Mアクセス回路113、C Mアクセス回路112、バッファ115を有する。そして、複数のMP部から各S Mセクタへのバス数より各S Mセクタから共有メモリ160a, 160bへのバス数の方を少なく、複数のMP部から各C Mセクタへのバス数より各C Mセクタからキャッシュメモリ170a, 170bへのバス数の方を少なくしている。



【特許請求の範囲】

【請求項1】 一つ以上の中央処理装置と一つ以上の記憶装置間でデータの入出力を制御する記憶制御装置であって、

一つ以上のプロセッサと、前記記憶装置のデータを一時的に格納するキャッシュメモリと、前記キャッシュメモリおよび前記記憶装置に関する制御情報を格納している共有メモリと、セレクトを備え、前記プロセッサは前記共有メモリおよび前記キャッシュメモリに前記セレクトを介してアクセス可能であり、

前記プロセッサと前記セレクト間と、前記セレクトと前記共有メモリ間と、前記セレクトと前記キャッシュメモリ間はアクセスバスにより接続されており、

前記セレクトと前記共有メモリを接続する前記アクセスバス数の合計、または前記セレクトと前記キャッシュメモリを接続する前記アクセスバス数の合計は、前記プロセッサと前記セレクトとを接続する前記アクセスバス数の合計よりも少ないことを特徴とする記憶制御装置。

【請求項2】 請求項1記載の記憶制御装置において、前記プロセッサは、2つ以上の前記セレクトに接続され、前記プロセッサから前記キャッシュメモリへのアクセスルートおよび前記プロセッサから前記共有メモリへのアクセスルートをそれぞれ複数有し、通常は全ての前記アクセスルートを使用して負荷を均衡させ、障害等により一部の前記アクセスルートが使用不能になった場合は、残りの前記アクセスルートを使用することにより処理を継続させることを特徴とする記憶制御装置。

【請求項3】 請求項1または請求項2記載の記憶制御装置において、

前記プロセッサは、アクセス対象となるキャッシュアドレス、および共有メモリアドレスに応じて、使用すべき前記アクセスルートを決定する手段を有することを特徴とする記憶制御装置。

【請求項4】 請求項1記載の記憶制御装置において、前記共有メモリはベアをなす2つの共有メモリからなり、該各共有メモリは内部にアクセス回路を有し、該ベアの共有メモリには共有メモリ間バスが設けられ、一方の共有メモリはマスタ、他方の共有メモリはスレーブに定められ、

前記セレクトは、前記プロセッサから前記共有メモリへのライト処理時に、前記プロセッサからアドレスとライトコマンドとライトデータを受領し、ベアを形成する前記共有メモリの両方に対してアドレスとライトコマンドとライトデータを送信する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、ライトデータを指定のアドレスへ書き込む手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスを送信する手段を有し、

前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスと前記セレクトか

ら受領したアドレスとを比較する手段と、該手段の一致出力に応じて前記ライトデータを指定のアドレスに書き込む手段を有し、ライトデータのアクセス順を保証し、前記セレクトは、前記プロセッサから前記共有メモリへのリード処理時に、前記プロセッサからアドレスとリードコマンドを受領し、ベアを形成する前記共有メモリの両方に対してアドレスとリードコマンドを送信する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、リードデータを指定のアドレスから読み出し前記セレクトへ転送する手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスを送信する手段を有し、

前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスと前記セレクトから受領したアドレスとを比較する手段と、該手段の一致出力に応じてリードデータを指定のアドレスから読み出し前記セレクトへ転送する手段を有し、

前記セレクトは、前記マスタおよびスレーブの共有メモリから受領したリードデータを比較する手段と、該手段の一致出力に応じて前記リードデータを前記プロセッサへ送信する手段を有することを特徴とする記憶制御装置。

【請求項5】 請求項1記載の記憶制御装置において、前記共有メモリはベアをなす2つの共有メモリからなり、該各共有メモリは内部にアクセス回路を有し、該ベアの共有メモリには共有メモリ間バスが設けられ、一方の共有メモリはマスタ、他方の共有メモリはスレーブに定められ、

前記セレクトは、前記プロセッサから前記共有メモリへのライト処理時に、前記プロセッサからアドレスとライトコマンドとライトデータを受領し、ベアを形成する前記共有メモリのうちのマスタとなる共有メモリに対してアドレスとライトコマンドとライトデータを送信する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、ライトデータを指定のアドレスへ書き込む手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスとライトコマンドとライトデータを送信する手段を有し、

前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスに前記ライトデータを書き込む手段と、前記マスタ側の共有メモリに対して書き込みの完了報告を送信する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、前記スレーブ側の共有メモリから完了報告を受領したときライト完了とする手段を有し、ライトデータのアクセス順を保証し、

前記セレクトは、前記プロセッサから前記共有メモリへのリード処理時に、前記プロセッサからアドレスとリー

ドコマンドを受領し、ペアを形成する前記共有メモリのうちマスタ側の共有メモリに対してアドレスとリードコマンドを送信する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、リードデータを指定のアドレスから読み出す手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスとリードコマンドを送信する手段を有し、

前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスからデータを読み出す手段と、前記マスタ側の共有メモリに該読み出したデータを転送する手段を有し、

前記マスタ側の共有メモリのアクセス回路は、前記マスタ側の共有メモリから読み出したリードデータと前記スレーブ側の共有メモリから受領したリードデータを比較する手段と、該手段の一致出力に応じて前記セクタへ前記リードデータの転送と終了報告の送信を行う手段を有することを特徴とする記憶制御装置。

【請求項6】 請求項1記載の記憶制御装置において、前記プロセッサは、前記中央処理装置から受領したライトデータを前記キャッシュメモリに二重化して格納するためのキャッシュアドレスaとキャッシュアドレスbを決定する手段と、該キャッシュアドレスa及びキャッシュアドレスbと、ライトコマンドと、ライトデータを前記セクタに送信する手段を有し、

前記セクタは、前記キャッシュアドレスa及びキャッシュアドレスbと、ライトコマンドと、ライトデータを受領し、前記キャッシュアドレスaに対応すキャッシュメモリAに対して前記キャッシュアドレスaとライトコマンドとライトデータを送信し、前記キャッシュアドレスbに対応すキャッシュメモリBに対して前記キャッシュアドレスbとライトコマンドとライトデータを送信する手段を有し、

前記各キャッシュメモリは、指定されたキャッシュアドレスにライトデータを格納することを特徴とする記憶制御装置。

【請求項7】 請求項1記載の記憶制御装置において、前記プロセッサは、キャッシュアドレスaのデータをキャッシュアドレスbにコピーするキャッシュ間コピーを実行するため、コピー元のキャッシュアドレスaから前記セクタを経由してデータを読み出す手段と、該読み出したデータをコピー先のキャッシュアドレスbに前記セクタを経由して書き込む手段を有することを特徴とする記憶制御装置。

【請求項8】 請求項1記載の記憶制御装置において、前記プロセッサは、キャッシュアドレスaのデータをキャッシュアドレスbにコピーするキャッシュ間コピーを実行するため、キャッシュアドレスaからキャッシュアドレスbへのコピーを指示するコマンドを前記セクタに対して指示する手段を有し、

前記セクタは、前記プロセッサからの指示に応じてキ

ャッシュアドレスaからデータを読み出し該読み出したデータをキャッシュアドレスbへ転送する手段を有することを特徴とする記憶制御装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明はメモリを共有するマルチプロセッサシステムに関し、特にマルチプロセッサ型の記憶制御装置に関する。

【0002】

【従来の技術】 近年の記憶装置システムでは、より高い性能を得るため、マルチプロセッサ構成をとるシステムが主流になっている。例えば「HITAC H-6581-C3形ディスク制御装置」に示されている従来技術では、中央処理装置と制御装置間のデータ転送を実行する複数のホスト側プロセッサと、記憶装置と制御装置間のデータ転送を実行する複数の記憶装置側プロセッサと、記憶装置のデータを一時的に格納するキャッシュメモリと、複数プロセッサ間の制御情報を格納する共有メモリを備え、キャッシュメモリおよび共有メモリは全プロセッサからアクセス可能な構成となっている。この従来システムでは、プロセッサと共有メモリ、およびプロセッサとキャッシュメモリ間は1:1に接続されている。一方、「HITAC H6591形ディスク制御装置」に示されている従来技術では、制御装置は複数のホスト側プロセッサと、複数の記憶装置側プロセッサと、キャッシュメモリと、共有メモリを備え、各プロセッサと共有メモリ間は制御バスで接続され、各プロセッサとキャッシュメモリ間はデータバスにより接続されている。さらに本従来技術では、信頼性確保のため共有メモリは二重化されていて、一方が閉塞してもシステムは正常動作するようになっている。共有メモリの二重化状態を保持するために、この従来システムでは、共有メモリに対するライトアクセスが発生すると、両方の共有メモリ回路がアクセスを受領して、同時に指定アドレスを更新する方式をとっている。一方、「HITAC H-6581-C3形ディスク制御装置」に開示されている従来技術では、このような制御は行っておらず、更新が発生した場合には、両方の共有メモリのアドレスを、プロセッサで移動するプログラムによって逐次的に更新していた。

【0003】

【発明が解決しようとする課題】 記憶装置システムに対する高性能化要求に対して、制御装置の大規模化や構成要素の高速化、例えばプロセッサ数やキャッシュ容量の増大、高性能プロセッサの適用、内部バス幅の拡大やバス転送能力の向上などで対応してきた。しかしながら、「HITAC H6591形ディスク制御装置」に開示されている従来技術では、内部バスの転送能力がシステムの大規模化および性能向上に追従するのが困難になりつつある。特に制御バスなどは一度の転送量が少ないために、プロトコルオーバーヘッドに占有時間の大半が使用

されてしまい、バスの持つ転送能力が活かせない。

【0004】そこで高いメモリアクセス性能を得るために、「HITAC H-6581-C3形ディスク制御装置」で示した従来システムのように、プロセッサとメモリ間を1:1に接続する方法が考えられる。しかしながら、搭載するプロセッサ数の増大に伴い、共有メモリおよびキャッシュメモリに接続するアクセスバス数もそれに比例して増えていく。現在のLSIのピン数は最大600ピン程度である。一方で、アクセスバス幅を制御線もあわせて20ビット程度、プロセッサ数を64プロセッサと仮定すると、共有メモリおよびキャッシュメモリへは合計1280ビットが入線するため、LSIのピン数が不足する。また、パッケージの大きさに対する制限から、パッケージ上のコネクタ数も上限があり、1280ビットの入線は不可能である。以上から、本発明が解決しようとする第一の課題は、LSIのピンネックおよびコネクタネックを回避して、かつ必要十分な性能を確保する記憶制御装置の内部構成とすることにある。

【0005】一方、二重化された共有メモリでは、各プロセッサのアクセス順序を保証する必要がある。この問題を例をあげて説明する。例えばプロセッサaとプロセッサbが同時に共有メモリの内容をほとんど同時に更新したものと仮定する。

【0006】もし、共有メモリAの内容が、まずプロセッサaにより更新され、共有メモリBの内容が、まずプロセッサbにより更新され、その後、共有メモリAの内容がプロセッサbにより更新され、共有メモリBの内容がプロセッサaにより更新されたものとする、最終的なメモリ内容は、共有メモリAはプロセッサbの更新内容であり、共有メモリBはプロセッサaの更新内容となり、両面の状態が不一致となる。「HITAC H-6581-C3形ディスク制御装置」では、上記アクセス順序を保証するための手段として、プログラマ的に各プロセッサの排他を取った後、両方の共有メモリの同一アドレスを順次更新する方式が提示されている。しかし、この方式では、一旦ロックをかけてからメモリを更新するために、性能的に問題がある。以上から、本発明が解決しようとする第二の課題は、共有メモリ間の二重状態を保持することにある。

【0007】

【課題を解決するための手段】上記課題を解決するため、本発明は、一つ以上の中央処理装置と一つ以上の記憶装置間でデータの入出力を制御する記憶制御装置であり、一つ以上のプロセッサと、前記記憶装置のデータを一時的に格納するキャッシュメモリと、前記キャッシュメモリおよび前記記憶装置に関する制御情報を格納している共有メモリと、セレクトを備え、前記プロセッサは前記共有メモリおよび前記キャッシュメモリに前記セレクトを介してアクセス可能であり、前記プロセッサと前記セレクト間と、前記セレクトと前記共有メモリ間と、

前記セレクトと前記キャッシュメモリ間はアクセスバスにより接続されており、前記セレクトと前記共有メモリを接続する前記アクセスバス数の合計、または前記セレクトと前記キャッシュメモリを接続する前記アクセスバス数の合計は、前記プロセッサと前記セレクトとを接続する前記アクセスバス数の合計よりも少ないようにしている。

【0008】また、前記プロセッサは、2つ以上の前記セレクトに接続され、前記プロセッサから前記キャッシュメモリへのアクセスルートおよび前記プロセッサから前記共有メモリへのアクセスルートをそれぞれ複数有し、通常は全ての前記アクセスルートを使用して負荷を均衡させ、障害等により一部の前記アクセスルートが使用不能になった場合は、残りの前記アクセスルートを使用することにより処理を継続させるようにしている。また、前記プロセッサは、アクセス対象となるキャッシュアドレス、および共有メモリアドレスに応じて、使用すべき前記アクセスルートを決定する手段を有するようにしている。

【0009】また、前記共有メモリはベアをなす2つの共有メモリからなり、該各共有メモリは内部にアクセス回路を有し、該ベアの共有メモリには共有メモリ間バスが設けられ、一方の共有メモリはマスタ、他方の共有メモリはスレーブに定められ、前記セレクトは、前記プロセッサから前記共有メモリへのライト処理時に、前記プロセッサからアドレスとライトコマンドとライトデータを受領し、ベアを形成する前記共有メモリの両方に対してアドレスとライトコマンドとライトデータを送信する手段を有し、前記マスタ側の共有メモリのアクセス回路は、ライトデータを指定のアドレスへ書き込む手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスを送信する手段を有し、前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスと前記セレクトから受領したアドレスとを比較する手段と、該手段の一致出力に応じて前記ライトデータを指定のアドレスに書き込む手段を有し、ライトデータのアクセス順序を保証し、前記セレクトは、前記プロセッサから前記共有メモリへのリード処理時に、前記プロセッサからアドレスとリードコマンドを受領し、ベアを形成する前記共有メモリの両方に対してアドレスとリードコマンドを送信する手段を有し、前記マスタ側の共有メモリのアクセス回路は、リードデータを指定のアドレスから読み出し前記セレクトへ転送する手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスを送信する手段を有し、前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスと前記セレクトから受領したアドレスとを比較する手段と、該手段の一致出力に応じてリードデータを指定のアドレスから読み出し前記セレクトへ転送する手段を有し、前記セレクト

は、前記マスタおよびスレーブの共有メモリから受領したリードデータを比較する手段と、該手段の一致出力に応じて前記リードデータを前記プロセッサへ送信する手段を有するようにしている。

【0010】また、前記共有メモリはベアをなす2つの共有メモリからなり、該各共有メモリは内部にアクセス回路を有し、該ベアの共有メモリには共有メモリ間バスが設けられ、一方の共有メモリはマスタ、他方の共有メモリはスレーブに定められ、前記セレクトは、前記プロセッサから前記共有メモリへのライト処理時に、前記プロセッサからアドレスとライトコマンドとライトデータを受領し、ベアを形成する前記共有メモリのうちのマスタとなる共有メモリに対してアドレスとライトコマンドとライトデータを送信する手段を有し、前記マスタ側の共有メモリのアクセス回路は、ライトデータを指定のアドレスへ書き込む手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスとライトコマンドとライトデータを送信する手段を有し、前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスに前記ライトデータを書き込む手段と、前記マスタ側の共有メモリに対して書き込みの完了報告を送信する手段を有し、前記マスタ側の共有メモリのアクセス回路は、前記スレーブ側の共有メモリから完了報告を受領したときライト完了とする手段を有し、ライトデータのアクセス順を保証し、前記セレクトは、前記プロセッサから前記共有メモリへのリード処理時に、前記プロセッサからアドレスとリードコマンドを受領し、ベアを形成する前記共有メモリのうちマスタ側の共有メモリに対してアドレスとリードコマンドを送信する手段を有し、前記マスタ側の共有メモリのアクセス回路は、リードデータを指定のアドレスから読み出す手段と、前記スレーブ側の共有メモリに前記共有メモリ間バスを介して前記アドレスとリードコマンドを送信する手段を有し、前記スレーブ側の共有メモリのアクセス回路は、前記共有メモリ間バスから受領したアドレスからデータを読み出す手段と、前記マスタ側の共有メモリに該読み出したデータを転送する手段を有し、前記マスタ側の共有メモリのアクセス回路は、前記マスタ側の共有メモリから読み出したリードデータと前記スレーブ側の共有メモリから受領したリードデータを比較する手段と、該手段の一致出力に応じて前記セレクトへ前記リードデータの転送と終了報告の送信を行う手段を有するようにしている。

【0011】また、前記プロセッサは、前記中央処理装置から受領したライトデータを前記キャッシュメモリに二重化して格納するためのキャッシュアドレスaとキャッシュアドレスbを決定する手段と、該キャッシュアドレスa及びキャッシュアドレスbと、ライトコマンドと、ライトデータを前記セレクトに送信する手段を有し、前記セレクトは、前記キャッシュアドレスa及びキ

ャッシュアドレスbと、ライトコマンドと、ライトデータを受領し、前記キャッシュアドレスaに対応すキャッシュメモリAに対して前記キャッシュアドレスaとライトコマンドとライトデータを送信し、前記キャッシュアドレスbに対応すキャッシュメモリBに対して前記キャッシュアドレスbとライトコマンドとライトデータを送信する手段を有し、前記各キャッシュメモリは、指定されたキャッシュアドレスにライトデータを格納するようにしている。

【0012】また、前記プロセッサは、キャッシュアドレスaのデータをキャッシュアドレスbにコピーするキャッシュ間コピーを実行するため、コピー元のキャッシュアドレスaから前記セレクトを経由してデータを読み出す手段と、該読み出したデータをコピー先のキャッシュアドレスbに前記セレクトを経由して書き込む手段を有するようにしている。

【0013】また、前記プロセッサは、キャッシュアドレスaのデータをキャッシュアドレスbにコピーするキャッシュ間コピーを実行するため、キャッシュアドレスaからキャッシュアドレスbへのコピーを指示するコマンドを前記セレクトに対して指示する手段を有し、前記セレクトは、前記プロセッサからの指示に応じてキャッシュアドレスaからデータを読み出し該読み出したデータをキャッシュアドレスbへ転送する手段を有するようにしている。

【0014】

【発明の実施の形態】図1に本発明における記憶制御装置の内部構成例を示す。制御装置195はMP (Micro Processor) 部110と共有メモリ (SM) セレクト140と、キャッシュ (CM) セレクト150と、共有メモリ (SM) 160と、キャッシュメモリ (CM) 170から構成される。MP部110は、1つ以上のプロセッサ (MP) 111と、CMアクセス回路112と、SMアクセス回路113を内部に持つ。MP111は、記憶装置とCM170間のデータ転送、あるいはCM170と中央処理装置間のデータ転送を実行する。

【0015】CMアクセス回路112は、MP111からの要求に応じてCM170と記憶装置、あるいはCM170と中央処理装置間のデータ転送を実行するDMA回路である。バッファ115は、CMアクセス回路112の指示により、データを一時的に格納するために用いられる。LM114は、ローカルメモリであり、プロセッサのワークで使用するメモリとして、また、リードデータ、ライトデータを格納するメモリ等として使用される。SMアクセス回路113は、MP111からの要求に応じてSM160からMP111へのデータ転送、あるいはMP111からSM160へのデータ転送を実行する。

【0016】SMセレクト140は、複数のMP111

からSM160へのアクセス要求のうち、一つを選択して実行する機能を持つ。CMセクタ150は、複数のMP111からCM170へのアクセス要求のうち、一つを選択して実行する機能を持つ。本実施例ではセクタをSMセクタとCMセクタに分離した構成としているが、このような分離をせずに1つのセクタとし、1つのセクタでSMセクタとCMセクタの機能を兼ね備える構成としてもよいことは言うまでもない。

【0017】SM160は、キャッシュ管理情報や、システム管理情報などの制御情報を格納しており、2つのSM160aおよびSM160bでペアを形成する。格納データはペア間で二重化されていて、同一アドレスには同一のデータが格納されている。一方が閉塞した場合でも、システムダウンに至ることはない。SM160aとSM160b間を接続する共有メモリ間バス165によって、格納データの二重化を実現する。本実施例では、SM160をCM170とは独立したメモリモジュールとして表現しているが、CM170の一部をSM160として使用する構成も考えられる。

【0018】一方、CM170は、記憶装置上のデータを一時的に格納するメモリである。ホストからのライトデータで、記憶装置に未反映のデータは、2つのCM170aとCM170bに二重化して格納されている。ここで、SM160の二重化と異なる点は、必ずしも同一のキャッシュアドレスに同一データが格納されていない点である。CM170の管理情報はSM160に格納されていて、MP111はSM160にアクセスしてCM170aとCM170bのデータ格納領域を確保し、当該領域にライトデータを書込むことにより、二重化を実現する。このため、SM間アクセスバス165のような機構は不要である。

【0019】MP部110からは2本のSMアクセスバス120によって相違なる2個のSMセクタ140に接続され、同様に2本のCMアクセスバス130によって相違なる2個のCMセクタ150に接続される。また、一つのSMセクタ140およびCMセクタ150へは、4個のMP部が接続される。SMセクタ140からはそれぞれ1本のSMアクセスバス125によってSM160aとSM160bに接続され、CMセクタ150からはそれぞれ1本のCMアクセスバス135によってCM170aとCM170bに接続される。このような4つのMP部110と2つのSMセクタ140と2つのCMセクタ150で一つのグループを形成し、セクタグループ190と呼ぶ。制御装置195は1個以上のセクタグループ190を有する。

【0020】ここで、上記個数は一実施例に過ぎず、個数を上記に制限するものではない。重要なことは、セクタグループ190内ではMP部110の個数よりもSMセクタ140の個数を少なくすることであり、これによってMP部110とSMセクタ140とを接続す

るSMアクセスバス120の個数よりも、SMセクタ140とSM160間を接続するSMアクセスバス125の個数を少なくでき、SM160に搭載されたLSIのピンネック、およびSM160のパッケージのコネクタネックを解消することができる。このことは、CM170についても同様である。例えば、32個のMP部110からSM160a、160bに直接に独立したSMアクセスバス120が2本ずつ（1本は交代バス用）存在した場合、SM160に接続するSMアクセスバス120の個数は、合計32(MP)×2(path)=64本となる。仮に各SMアクセスバス120が16ビット幅とすれば、1024ビットのピンがSM160に必要なになるが、現在このようなピン数を持つLSIは存在しない。一方、本実施例によると、32個のMP部110に対しては8個のセクタグループ190を有し、SM160に接続するSMアクセスバス120の個数は、8(セクタグループ)×2(SMセクタ)=16本に抑えられ、ピン数は256本で済む。

【0021】次に、SMアクセスのための実施例である実施例1について説明する。まず、図2から図6を使用して、SMアクセス時の処理を説明する。図2は、本実施例におけるSM160のアクセスシーケンス例を示す。MP111からSMアクセス回路113へは、アクセスするSM160のアドレスと、リードの場合はデータを格納するLM（ローカルメモリ）114のアドレス、ライトの場合はライトデータを格納しているLM114のアドレスを渡す。SMアクセス回路113がMP111よりリードアクセス要求を受領すると、SMセクタ140に対してREQ信号を送信し、SMセクタ140からACK信号が帰るまで待つ。その後、リードコマンドとリードアドレスを送信する。SMセクタ140では、複数あるSMアクセス要求のうち、一つを選択して、ACK信号を返す。その後コマンドとアドレスを受信すると、二重アクセスの場合には、SM160aおよびSM160bに対して、コマンドとアドレスを送信する。SM160は、コマンドとアドレスを受信すると、当該アドレスからデータを読み出し、SMセクタにリードデータを転送した後、終了報告をSMセクタ140に送信する。SMセクタ160は、最初のリードデータと終了報告をSM160aまたはSM160bから受信すると、当該データおよび終了報告をMP部110へ送信する。この際、リードデータは、SMセクタ140のバッファへ一時的に格納しておく。その後、他のSM160aまたは160bからリードデータを受信すると、バッファへ格納しておいたリードデータとコンペアを実施して、一致していることを確認する。続いて受信した終了報告をMP部110へ転送し、処理を完了する。

【0022】SMアクセス回路113は、リードデータを受信すると、MP111から指定されたLM114の

アドレスへ当該データを格納する。その後、二つの終了報告を受信して、処理完了となる。ライト処理の場合もほぼ同様で、SMアクセス回路113は、コマンド、アドレスに続いてライトデータを送信する。SMセクタ140もコマンド、アドレス、ライトデータをSM160aおよびSM160bに転送し、SM160aおよびSM160bは、指定されたアドレスにライトデータを格納した後、終了報告を行う。

【0023】図3はSMセクタ140の内部構成図を示す。各MP部110とSMセクタ140を接続するSMアクセスバス120毎にレジスタ310が対応していて、制御部300は、このレジスタ310をリード・ライトすることによって、MP部からの情報を受信できる。一方、SM160aおよびSM160bとSMセクタ140を接続するSMアクセスバス125毎にレジスタ340aおよびレジスタ340bが対応していて、このレジスタ340をリードライトすることにより、SM160と通信することが可能である。SM160からリードしたデータは、アクセス要求のあったMP部110へ、レジスタ310経由で転送すると同時に、バッファ330へ格納することが可能である。そして、SM160aとSM160bの両方からデータをリードした後、データコンペア部320で二重化のチェックを行う。制御部300は、MP部110からのアクセス要求を選択して実行する機能と、データコンペア部320の制御機能などを有する。

【0024】図4は、SMアクセス回路113の処理フローを示す。

【0025】SMアクセス回路113がプロセッサ111からSMアクセス要求を受領すると、ステップ400でアクセスするアドレスのビット8が0かどうか調べ、もし0であればステップ410でSMセクタ140aに対してリクエスト信号を送信する。もし1であればステップ420でSMセクタ140bに対してリクエスト信号を送信する。MP部は2つのSMセクタ140aおよび140bの両方に対してSMアクセスバス120を持っているので、両方のバスを負荷分散させたほうが効率がよい。このため本実施例では、SM上のアドレスを128バイト刻みでSMセクタ140aと140bを切り替える。よって、アクセスアドレスの8ビット目を見て、どちらを選択するか決定すれば良い。一般的には、第nビット目を見てアクセスバスを選択することにより、2(n-1)バイト刻みでSMセクタ140aと140bを切り替えることになる。このnは、システムのSMアクセスパターンなどを調査して決めるのが良い。

【0026】送信したREQ信号に対して、ステップ411または421でSMセクタ140からACK信号が返されると、ステップ412、413または422、423で、SMセクタ140に対してコマンド、アクセス

アドレスを送信する。続いてアクセスがリードの場合は、ステップ414または424でSM160からデータを受領し、プロセッサから指定されたLM114のアドレスに当該データを格納する。もしアクセスがライトの場合は、ステップ413、414またはステップ423、424に続いて、プロセッサ111から指定されたLM114のアドレスからライトデータを送信する。最後にステップ415でSMセクタ140からSM160aの終了報告とSM160bの終了報告を受信したら処理を完了する。

【0027】ここで、ステップ410、411またはステップ420、421でREQ-ACKのやりとりを必要とする理由を説明する。SMセクタ140は、後に詳しく説明するが、複数のMPからのアクセス要求を順次実行するために、アクセス要求がぶつかった場合は待たなくてはならない。そこで、SMアクセス回路113はまずリクエスト信号を送信し、これに対してSMセクタからACK信号が返ると、処理開始となる。SMアクセス回路113からSMセクタ140へ送信されるコマンドデータには、リードあるいはライトを示す情報と、二重アクセスか一重アクセスかを示すフラグと、一重アクセスの場合はSM160aかSM160bのいずれへアクセスするかを示す情報を含む。

【0028】次に図5および図6を用いて制御部300の処理を説明する。制御部300はステップ510および511において、レジスタ310を0番から7番まで、REQ信号がオンになっているかどうかをラウンドロビンで監視する。オンになっているレジスタ310を見つけると、当該バスに対してACK信号を返し、処理可能となったことを知らせる。これは当該レジスタ310のACKに対応するビットをオンにすることによって成される。続いてステップ512でSMアクセス回路113から受信したコマンドとアドレスをSM A面とB面の両方に転送する。

【0029】ここでコマンドをデコードして、もし二重アクセスのリードコマンドであれば、ステップ514でSM160から、リードデータが転送されてくるのを待つ。二重アクセスの場合は、SM A面160aから読み出されたデータと、SM B面160bから読み出されたデータが一致することを確認する必要がある。また、SM A面とB面のアクセス処理はそれぞれ独立して行われるので、データが読み出される順序は決まっていない。そこで、ステップ514でリードデータを受信すると、ステップ515で他の面からのデータが受信済みであるかどうかをチェックする。受信済みであれば、ステップ516で、受信データをバッファ330へ格納する。また、最初の受信データはMPへ転送する。もし今受信したデータがSM A面160aからのものであれば330aへ、SM B面160bからのものであれば330bへ格納した後、データコンペア部320に対してデータコンペ

アを指示する。もし、ステップ518でコンペア結果に異常がなく、かつステップ517でSM160から正常終了が報告されていればアクセス成功とし、ステップ519で正常終了をSMアクセス回路113に報告して、処理を終了する。

【0030】次にステップ515で他の面からリードデータを受信していなかった場合、つまり今受信したデータが最初にSM160から転送されたデータだった場合は、ステップ520で、もしSM A面160aからのデータであればA面バッファ330aへ、B面160bからのデータであればB面バッファ330bへ、受信データを格納する。また、バッファ330へ格納すると共に、当該リードデータをSMアクセス回路113へも転送する。その後ステップ521および522で、SM160から受信した終了報告をSMアクセス回路113へ転送する。データをバッファ330へ格納しておくのは、後のコンペアのためにデータを保持しておくためである。また、最初のリードデータをSM160から受信した時点で、データをSMアクセス回路113に転送し、後のリードデータは転送しない。これは、データ転送を一回にすることにより、無駄な転送オーバーヘッドを削減するためである。ステップ517またはステップ521で、SMより正常終了を受信しなかった場合には、ステップ523で異常終了をMPに報告する。

【0031】ステップ513で、SMアクセス回路113からの要求がライトアクセスか、または一重リードアクセス時の処理を図6に示す。ライトアクセス要求の場合は、ステップ620でSMアクセス回路113から受信したライトデータをSM A面160aおよびSM B面160bへ転送し、ステップ621で両面から完了報告を受領したら、それぞれをSMアクセス回路113へ転送する。また、一重リード要求の場合は、ステップ611で指定された面からリードデータを受領すると、それをSMアクセス回路113へ転送し、その後ステップ612で指定された面からの完了報告を受領したら、それをSMアクセス回路113へ転送して処理を完了する。一重リードもライトアクセスも、SM A面160a、SM B面160bの両方へコマンドおよびアドレスを転送しているが、SM160側ではコマンドをデコードした時に、自SM160に対するアクセスかどうかを判断し、処理不要の場合は要求を無視する。こうすることによって、SMセクタでは一重アクセスかどうか判断する必要がなく、負荷を低減することが可能である。

【0032】SM160へのアクセスは、ペアとなるSM160の両方にアクセスする二重アクセスと、いずれか一方にアクセスする一重アクセスの二種類のアクセスモードがある。これには、制御情報として、一重で格納されている情報と二重で格納されている情報があるからで、前者の例としては、キャッシュの管理情報、後者の例としては、システム管理情報などがある。

【0033】SM160は、二重アクセスに対してはそのアクセス順を守る必要がある。まずその理由について、例を用いて説明する。今、プロセッサ#1がSM160の内容をAに更新しようとしていて、同時にプロセッサ#2が同一アドレスの内容をBに更新しようとしているとする。もしSM A面160aがプロセッサ#1、プロセッサ#2の順にアクセスを実行し、SM B面160bがプロセッサ#2、プロセッサ#1の順にアクセスを実行したとすると、最終的に当該アドレスの内容は、SM A面160aはBに、SM B面160bはAに更新され、データの不一致を引き起こす。さらに他の例で説明する。今SM160の内容がAであるとし、プロセッサ#1がSM160の内容を読み出そうとしていて、同時にプロセッサ#2が同一アドレスの内容をBに更新しようとしているとする。もしSM A面160aがプロセッサ#1、プロセッサ#2の順にアクセスを実行し、SM B面160bがプロセッサ#2、プロセッサ#1の順にアクセスを実行したとすると、SM A面からは更新前のデータ、すなわちAが読み出され、SM B面からは更新後のデータ、すなわちBが読み出され、データの不一致を検出する。以上の例からわかるように、SM160の二重状態を保持するための制御が必要である。

【0034】上記二重状態を保持する制御の基本的考え方は、SM160の一方をマスタ、他方をスレーブと定めて、マスタ側がアクセス実行を許可するまでは、スレーブ側は当該アクセスを実行しない、ということである。マスタおよびスレーブの切り替えは、面単位、例えばSM160aを常にマスタとし、SM160bを常にスレーブとして取り扱う方法や、アドレス領域毎に取り扱う方法、例えばSM160aとSM160bを256バイト毎にマスタおよびスレーブを切り替える方法などが考えられる。本実施例では、後者、すなわちアドレス単位毎にマスタとスレーブを切り替える方法を前提とする。この場合、例えばアドレス単位を保持するレジスタをSM160およびSMセクタ140内部に保持して、システム立ち上げ時にMPが設定するようにしておき、SMアクセスが発生した場合に、SM160およびセクタ140は、アクセスアドレスとレジスタに格納されたアドレス単位を比較して、いずれがマスタになるかを調べれば良い。

【0035】図7はSM160の構成図を示す。SM160は、SMセクタ140とSM間の情報の送受信を行うMPインタフェース710、二重化の他面へ情報を送信するSM送信インタフェース720、他面から情報を受信するSM受信インタフェース730、データを格納するメモリバンク750、メモリバンクからのリードライトを制御するメモリコントローラ740、MPインタフェース710とSM送信インタフェース720とSM受信インタフェース730とメモリコントローラを制御するSMコントローラ700から構成される。

【0036】図8にSMコントローラ700の処理フローを示す。まずステップ800で、SM受信インタフェース730からアクセス対象となるアドレスを取得する。もしNULLアドレスであれば、ステップ810でMP111からのアクセス要求が存在するかどうか、MPインタフェース710を順に調べていく。もしアクセス要求があれば、MPインタフェース710はコマンド、アドレス、およびライトの場合はライトデータを保持している。そこでまずアドレスを取得して、自面がマスタとなって処理すべきアドレスかどうかを調べる。自面がマスタとなるべきアクセス要求が存在しなければ、ステップ800に戻り、ポーリングを継続する。

【0037】自面がマスタとして処理するアクセス要求が存在すれば、ステップ811でアクセス要求が二重アクセスかどうかをコマンドから調べる。もし二重アクセスであればステップ812でSM送信インタフェース720に当該アドレスを送信して、スレーブ面に対して当該アクセスの処理を要求する。続いてステップ813で、当該アドレスにアクセスする。もしリード要求であれば、メモリコントローラ740に対してリード要求とアドレスを送信する。メモリコントローラ740がメモリバンク750にアクセスしてデータが読み出されると、SMコントローラ700は当該データをMPインタフェース710に転送して処理を完了する。もしライトアクセスであれば、メモリコントローラ740に対してライト要求を送信するとともにアドレスおよびライトデータを転送する。メモリコントローラ740は指定アドレスにデータを書き込み処理を完了する。

【0038】以上の処理が正常に完了すると、SMコントローラ700はMPインタフェース710を介してMP111に正常終了を報告し、ステップ800に戻る。

【0039】次にステップ800でSM受信インタフェース730に非ヌルアドレスが格納されていた場合、すなわちマスタ面からの処理要求があった場合は、ステップ820で指定アドレスと一致するアクセス要求があるかどうかをMPインタフェース710を走査する。一致するアドレスが存在した場合はステップ813で、当該アドレスへのアクセスを実行する。もし一致するアドレスが存在しなかった場合は、ステップ830で同一のアクセス要求があるまで、MPインタフェース710を順に監視し続ける。もし、一定時間待ってもアクセス要求がなかった場合は、タイムアウトエラーとしてMP111に報告する。

【0040】ステップ820では、SM受信インタフェース730から受信したアドレスと一致するものを、MPインタフェース710を順に探し回るのは、これは多少効率が悪い。そこで、マスタ面から要求があったら、どのMPインタフェース710に対応する要求か判断できると都合が良い。このため、例えばシステムに一意に付けられているMP番号をコマンドに格納しておき、S

M送信インタフェース720にはMP番号を送信する。さらに、各SMセクタ140が接続する2つのMPインタフェース710を固定的に決めておき、セクタはMP番号と、対応するMPインタフェースの対応表を持っておく。このようにすると、スレーブ側でSM受信インタフェース730から受信したMP番号から、MPインタフェース710の番号が一意に定まり、すぐさまステップ820からステップ813へ移行できる。

【0041】以上の方式によると、SM送信インタフェース620およびSM受信インタフェース630を介した制御により、二重アクセスにおいては、マスタ面が処理を開始しない限りはスレーブ面も当該アドレスの処理は行わないために、アクセス順序は必ず守られ、前述したようなデータの不一致は起こり得ない。

【0042】次に、SMアクセスの実施例である実施例2について説明する。実施例2では、SM160への二重アクセスにおけるアクセス順序の保証方法として第2の保証方法を用いている。実施例2について、図9から図11を用いて説明する。図9はリードアクセスにおけるSMアクセスプロトコルの例を示す。実施例2の第二の保証方法では、SMセクタ140は、マスタとなるSM160へのみ、アクセス要求を送信する。ここでは、SM160aがマスタ、SM160bがスレーブになっているとする。リードアクセスの場合は、SM160aがアクセス要求を受領すると、SM間アクセスバス165を介して、SM160bにコマンドとアドレスを送信するとともに、メモリをアクセスしてデータを読み出す。SM160bは、このアクセス要求に対してメモリからデータを読み出し、SM160aにデータを送信した後、SM160aに対して終了報告を行う。SM160aはSM160bからデータおよび終了報告を受領して、正常にデータが読み出されたことを確認した後、自メモリから読み出したデータと、SM160bから受信したデータをコンペアして、一致していればSMセクタ140にデータを送信し、終了報告を行う。ライトアクセスの場合は、SM160aはSMセクタ140からコマンド、アドレスに続いてライトデータを受信するが、SM160bに対しても、コマンド、アドレス、データを送信するとともに、メモリの指定アドレスにデータを格納する。一方、SM160bでは、SM160aから指定されたアドレスにデータを格納した後、SM160aに対して終了報告を行う。SM160aは、自メモリへの書き込みが正常に終了し、かつSM160bからの終了報告を受領するのを待って、SMセクタ140に終了報告する。

【0043】SMセクタ140は、MP部からアクセス要求を受けると、マスタとなるSM160へのみアクセス要求を転送すればよい。また、データコンペアや二重アクセスの待ち合わせなどの処理はSM160側で行う。

【0044】図10にSMコントローラ700の処理を示す。SMコントローラ700は、ステップ1000でマスタ面からのアクセス要求の有無をSM受信インタフェース730をアクセスして調べる。もし、アクセス要求があれば、SM受信インタフェース730には、コマンド、アドレス、そしてライトの場合はライトデータが格納されている。もしアクセス要求がなかった場合には、ステップ1010で、MPインタフェース710を順にアクセスしてアドレスを取得し、自面がマスタとなるべきアクセス要求があるかどうかを判定する。もしなければ、ステップ1000に戻り、ポーリングを繰り返す。

【0045】もし自面がマスタとなるべきアクセス要求があれば、ステップ1011でコマンドをデコードして二重アクセスかどうかを調べ、もし二重アクセスであれば、ステップ1012で、SM送信インタフェース720へ、アクセス要求、すなわちコマンド、アドレス、そしてライトの場合はライトデータを転送する。さらにリードアクセス要求であれば、ステップ1014で自メモリからデータを読み出した後、スレーブからのデータ転送および終了報告を待つ。終了報告があると、ステップ1015で、自メモリから読み出したリードデータと、スレーブから転送されたリードデータをコンペアして、一致していれば、MPインタフェース710ならびにSMセクタ140経由でリードデータを転送し、その後、終了報告を行う。もし、データ不一致となるか、あるいはマスタ面からのデータ読み出しに失敗した場合、または、スレーブ面から異常終了が報告された場合は、MP111に対して異常終了を報告する。

【0046】ステップ1013で、ライトアクセス要求だった場合は、ステップ1030で自メモリの指定アドレスへライトデータを格納した後、スレーブからの終了報告を待つ。その後、ステップ1031で、自メモリへのアクセスが正常終了し、かつスレーブから正常終了の報告があれば、MPインタフェース710およびSMセクタ140経由で、MP111に対して終了報告を行う。もし、自メモリへのアクセスが異常終了するか、あるいはスレーブから異常終了が報告された場合は、MP111に対して、異常終了を報告し、処理を完了する。

【0047】ステップ1011で、一重アクセス要求の場合は、上記で説明してきた二重アクセスのための制御は必要ない。よって、自メモリへアクセスを実行して、MP111に終了報告を行った後、処理を完了する。

【0048】ステップ1000において、マスタ面からのアクセス要求が存在した場合、それがリードアクセスであれば、図11のステップ1110で、自メモリからデータを読み出し、マスタへデータを転送し、終了報告を行う。もし、ライトアクセスであれば、ステップ1120で、自メモリへデータを格納した後、マスタへ終了報告を行い処理を完了する。

【0049】上記実施例2の第二の保証方法によると、スレーブは、マスタからのアクセス要求を受けてから当該処理を実行する。このため、アクセスの順序は必ず保証される。

【0050】実施例1では、マスタとスレーブのアクセス完了の待ち合わせをSMセクタ140で行っている。SM160aとSM160bは、自メモリアccessが完了すると、メモリを解放することができた。一方、実施例2では、マスタ側でスレーブのアクセス完了を待つために、それだけ長時間SM160が占有されることになる。このため、性能的には、実施例1のほうが優れている。

【0051】次に本発明におけるキャッシュメモリ(CM)170へのアクセス方法について説明する。CM170は、記憶装置上のデータを一時的に格納しておくためのメモリであり、ダーティデータ、すなわちホストからのライトデータで、CM170には格納しているが、記憶装置へは未反映のデータは二重化されてCM170に格納されている。この二重化データは、SM160の二重化とは異なり、MP111で移動するプログラムが、CM170aおよびCM170bに領域を確保して、当該領域に対してデータを格納するために、相異なる領域に二重化データを格納することになる。また、SM160に対するリードアクセスのようなコンペアチェックは行わない。

【0052】CMセクタ150の構成は、図3で示したSMセクタ140の構成とほぼ同じである。ただし、前述したように、キャッシュデータのコンペアは行わないので、データコンペア部320、バッファ330は持たない。また、CM170の構成は、図7で示したSM160の構成とほぼ同じである。ただし、前述したように、同一アドレスに同一データを二重化することではなく、アクセス順序を保証する必要もないため、SM送信インタフェース720、SM受信インタフェース730に相当するものは持たない。

【0053】以下、CM170へのライトデータの格納方法についての実施例である実施例3について説明する。なお、MP111は、ライトデータを格納すべきCM領域はすでに取得済みであるとする。MP111は、CMアクセス回路112に、コマンド、データを格納すべきCMアドレスaおよびCMアドレスbを設定し、CMアクセス回路112を起動する。ここで、CMアドレスa、CMアドレスbは、それぞれCM170a、CM170b上のアドレスである。CMアクセス回路112が起動されると、データは一定の大きさのバケットに分割されて、順次転送される。それぞれのバケットには、コマンド、およびアドレスが付与され、データ転送の実行に伴い、それぞれCMアクセス回路112にて更新される。例えば、アドレスaから24KBのデータを2KBのバケットで転送する場合、最初のバケットには、ライ

トコマンドおよびアドレスaが、次のバケットにはライトコマンドおよびアドレスa+2048が、第n番目のバケットには、ライトコマンドおよびアドレスa+2048*(n-1)が付与され、合計12個のバケットにより全データがCM170へ転送されることになる。

【0054】以下、図12を用いてキャッシュに対する二重ライトについて説明する。本実施例では、まず片方の面(図12の例ではCM170a)にライトした後、他方の面(図12の例ではCM170b)へライトする。従って、MP111は、まずCM170aに対するDMAリスト、すなわち、コマンド、アドレス、転送長等から構成され、CMアクセス回路112の処理内容が記述されているリストと、CM170bに対するDMAリストを作成した後、CMアクセス回路を起動する。

【0055】CMアクセス回路112は、転送すべきアドレス毎にどのMP-CM間アクセスバス130を使用するか選択する。選択方法は、SMアクセス回路113で説明した方法と同様に、例えば32KB毎に、使用するMP-CM間アクセスバス130を切り替えれば良い。MP-CM間バスを選択すると、CMセクタ150に対して、REQ信号と最初のバケットのコマンドを送信する。コマンドには、リード・ライト種別と、二重アクセスかどうかを示すフラグと、a面への転送を示すフラグとb面への転送を示すフラグから成る。

【0056】CMセクタ150はコマンドから、CM170aへのライトであることがわかると、CM170aに対してREQ信号とコマンドを送信する。CM170aが処理可能になり、ACK信号を受信すると、CMアクセス回路113にACKを送信する。

【0057】ここで、CM170に対してREQ-ACKのプロトコルが必要な理由について説明しておく。もし、CM170にデータを受信するのに十分な大きさのバッファを設けていれば、REQ-ACKプロトコルは必要でなく、データをバッファにため込んでおき、後にCM170のコントローラが処理可能になった時点でバッファからキャッシュメモリへライトすれば良い。しかし、一般にキャッシュアクセスの場合は転送するデータ長が大きく(本実施例では2KBを仮定している)、各キャッシュアクセスバス130毎にこのようなバッファを準備するのはコスト的に無駄がある。そこで、CM170が処理可能になった時点で、CMセクタ150に対してACKを返し、転送されたデータをスルーでキャッシュメモリへ転送することにより、上記のようなバッファを不要としている。SMアクセスの場合は、一回の転送が1ワード程度と小さいため、SMアクセスバス120毎にバッファを準備していても、たいした問題とならない。このため、SMセクタ140からSM160へはREQ-ACKプロトコルは必要とせず、レジスタ310にコマンド、アドレス、データを格納することとした。

【0058】さて、CMアクセス回路112のACK受

領すると、アドレス、データを転送して、終了報告を待つ。CMセクタ回路では、アドレス、コマンドをアクセス対象となるCM170aへ転送し、CM170aがデータの書き込みを完了し、終了報告を受け取ると、これをCMアクセス回路112へ転送して処理を完了する。その後CMアクセス回路112は、CM170bに対してライト処理を実行するが、シーケンスについてはCM170aへのライト方法と同様なので省略する。以上でCM170a、CM170bへの1バケット分の二重ライトが完了するが、この処理を全バケット分繰返す。

【0059】上記実施例3では、CM170aとCM170b対応にコマンド、アドレス、データを転送しているため、MP-CM間アクセスバス130、135ならびにCMセクタ150の利用効率が悪い。そこで、この問題を解決するための実施例である実施例4を以下に説明する。図13は、CM170に対する二重ライトのシーケンスを示す。CMアクセス回路112はCMセクタ150に対して、REQ信号とともに、コマンドを転送する。ここでコマンドは、ライトアクセスコマンドであり、二重アクセスフラグと、CM170aへのアクセスフラグと、CM170bへのアクセスフラグがオンになっている。CMセクタ150は、コマンドを受領すると、CM170aとCM170bの両方に対してREQ信号とコマンドを転送する。CM170aとCM170bの両方が処理可能な状態となり、ACK信号が返されると、CMアクセス回路112へACK信号を送信する。その後、CMアクセス回路112はCM170aのアドレス、CM170bのアドレス、ライトデータを転送し、これを受けてCMセクタ150は、CM170aのアドレスはCM170aへ、CM170bのアドレスはCM170bへ転送する。そして、ライトデータは、CM170aとCM170bの両方へ転送する。CM170は上記実施例3と同様に、指定アドレスへライトデータを格納した後、終了報告をCMセクタ150経由でCMアクセス回路112へ行う。上記実施例4によると、ライトデータの転送は1回でよく、その分MP-CMセクタ間アクセスバス130の占有時間は短くて済む。また、CM170aとCM170bの処理は並行に行われるので、レスポンスも短縮できる。

【0060】次に実施例5として、キャッシュ間でデータのコピーを行う方法について述べる。キャッシュ間コピーでは、コピー元およびコピー先がそれぞれ、CM170aからCM170aへ、CM170bからCM170bへ、CM170aからCM170bへ、CM170bからCM170aへ、の4通りが考えられる。まず最初に、上記4通りの全てに適用可能な手法、すなわち、MP部のバッファ115を使用して、コピー元のCM170からバッファ115へデータを読み出した後、バッファ115からコピー先のCM170へデータをコピー

する方法を以下に説明する。

【0061】図14は、バッファ115介したキャッシュ間コピー方法のシーケンスをまとめた図である。バッファ115を介した場合は、バッファ115の大きさを超えるコピーは一度にできないため、複数回に分ける必要がある。この制御はMP111で行う。MP111は、まず、コピー元のCM170（図14の場合はCM170a）に対するリード要求のDMAリストと、コピー先のCM170（図14の場合はCM170b）に対するライト要求のDMAリストを作成し、CMアクセス回路112を起動する。ここで、DMAリスト中の転送サイズの上限はバッファサイズである。CMアクセス回路は、これまで説明してきた方法により、まずコピー元のCM170aからバッファ115へデータを読み出す。これが完了すると、続いてコピー先のCM170bへバッファ115からデータを書込み、処理を完了し、MP111にコピー完了を通知する。コピー開始を指示してから完了報告を受領するまで、MP111が介入することはない。もし、バッファサイズの制限により、まだコピーすべきデータが残っている場合は、MP111は引き続きDMAリストを作成し、コピー処理を続行する。

【0062】次に実施例6として、バッファ115を介しないキャッシュ間コピー方法について説明する。この方法は、コピー元から読み出したデータをCMセクタ150で折り返してコピー先へ転送するものである。データはCMセクタ150を（バッファなどに格納されず）スルーでコピー先へ流れていくので、コピー先とコピー元のキャッシュは異なっていなくてはならない。すなわち、本手法が適用できるのは、CM170aからCM170bへ、またはCM170bからCM170aへのいずれかである。

【0063】図15は、セクタで折り返すキャッシュ間コピー方法のシーケンスをまとめた図である。MP111は、キャッシュ間コピーを実行するDMAリストを作成したのち、CMセクタ150をREQ信号をアサートすることにより、起動する。ここで、リスト中には、キャッシュ間コピーコマンド、コピー元アドレス、コピー先アドレスを含む。CMセクタ150が起動されると、コピー元（図15の場合はCM170a）とコピー先（図15の場合はCM170b）に対してリクエスト要求を発行する。CM170aとCM170bの両方が処理可能になり、ACKが返されると、CMアクセス回路113に対してACKを返す。

【0064】これを受けてCMアクセス回路113は、コマンド、CM170aのアドレス、CM170bのアドレスを送信する。

【0065】CMセクタ150は、コマンドはCM170aとCM170bの両方に転送し、CM170aアドレスはCM170aに、CM170bアドレスはCM

170bに転送する。コピー元であるCM170aからデータが読み出されると、そのデータをコピー先であるCM170bへ転送する。CM170bの処理では、ライト処理と同様であり、すでにデータを受け取る準備ができているので、受領したライトデータを指定されたアドレスへ格納する。以上のCM170aからのリード処理と、CM170bへのライト処理を、全データのコピーを完了するまで繰り返す。

【0066】二つのキャッシュ間コピー方法の特徴を以下にまとめる。CMセクタ150で折り返すコピー方法では、コピー元からのリードとコピー先へのライトを並行して行える。また、コピー元から読み出したデータをバッファに溜めないため、どのようなデータサイズのコピーでも、一度のコピー起動で完了する。よってバッファ115を介してコピーする方法に比べて、コピーにかかる時間が短い。一方、コピーデータをバッファに溜めずにスルーでコピー先に転送するために、同一面、すなわちCM170aからCM170aへ、またはCM170bからCM170bへのコピーはできないデメリットがある。従って、同一面間のコピーはバッファ115を介したコピー方法で、別面間のコピーはCMセクタ150で折り返すコピー方法で行うのが良い。

【0067】

【発明の効果】本発明によると、複数のプロセッサとメモリの間にセクタを設置し、セクタでメモリへのアクセス要求を選択して実行することにより、パッケージ当たりの配線数を削減し、LSIのピンネック、およびパッケージコネクタネックを解消できる。さらに本発明によると、二重化された共有メモリの一方をマスタ、一方をスレーブと定め、共有メモリ間を接続して、スレーブ側はマスタ側の処理要求があった時のみ、二重アクセス要求を実行する機構を設けることにより、共有メモリ間の二重状態を保つことができる。

【図面の簡単な説明】

【図1】本発明における記憶制御装置の構成を示す図である。

【図2】実施例1のSMアクセスのシーケンスを示す図である。

【図3】SMセクタ回路の構成を示す図である。

【図4】SMアクセス回路の動作フローを示す図である。

【図5】SMセクタ回路の動作フローを示す図である。

【図6】図5に続くSMセクタ回路の動作フローを示す図である。

【図7】共有メモリの構成を示す図である。

【図8】実施例1のSMコントローラの動作フローを示す図である。

【図9】実施例2のSMアクセスのシーケンスを示す図である。

【図10】実施例2のSMコントローラの動作フローを示す図である。

【図11】図10に続くSMコントローラの動作フローを示す図である。

【図12】実施例3のCMアクセスのライトシーケンスを示す図である。

【図13】実施例4のCMアクセスのライトシーケンスを示す図である。

【図14】実施例5のCMアクセスのコピーシーケンスを示す図である。

【図15】実施例6のCMアクセスのコピーシーケンスを示す図である。

【符号の説明】

111 プロセッサ

112 CMアクセス回路

113 SMアクセス回路

114 ローカルメモリ

115 バッファ

120 SMアクセスバス

130 CMアクセスバス

125 SMアクセスバス

135 CMアクセスバス

140 SMセクタ

150 CMセクタ

160 共有メモリ

165 共有メモリ間バス

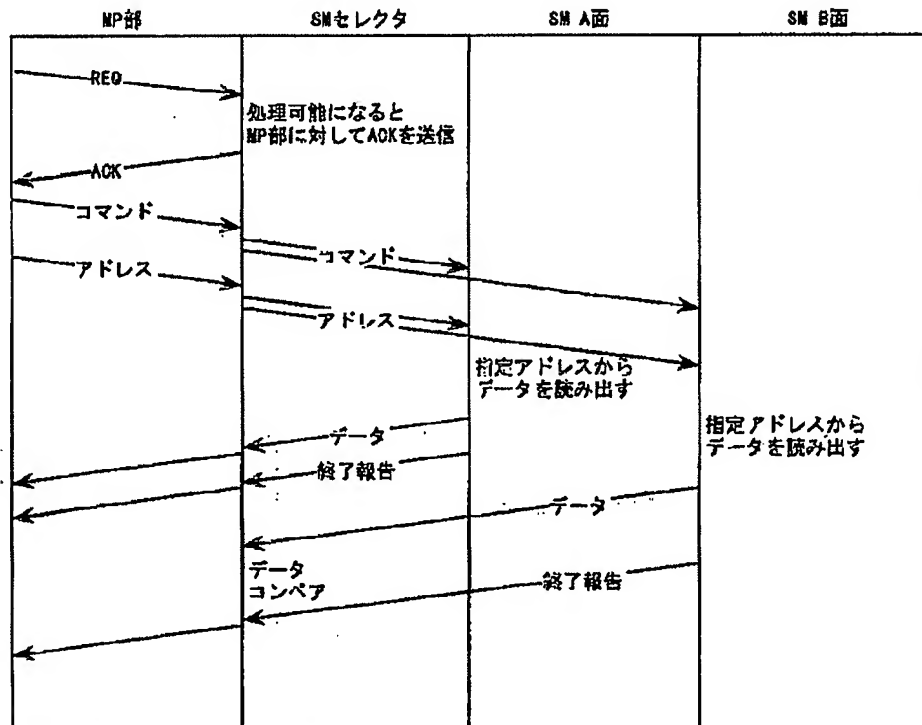
170 キャッシュメモリ

190 セクタグループ

195 記憶制御装置

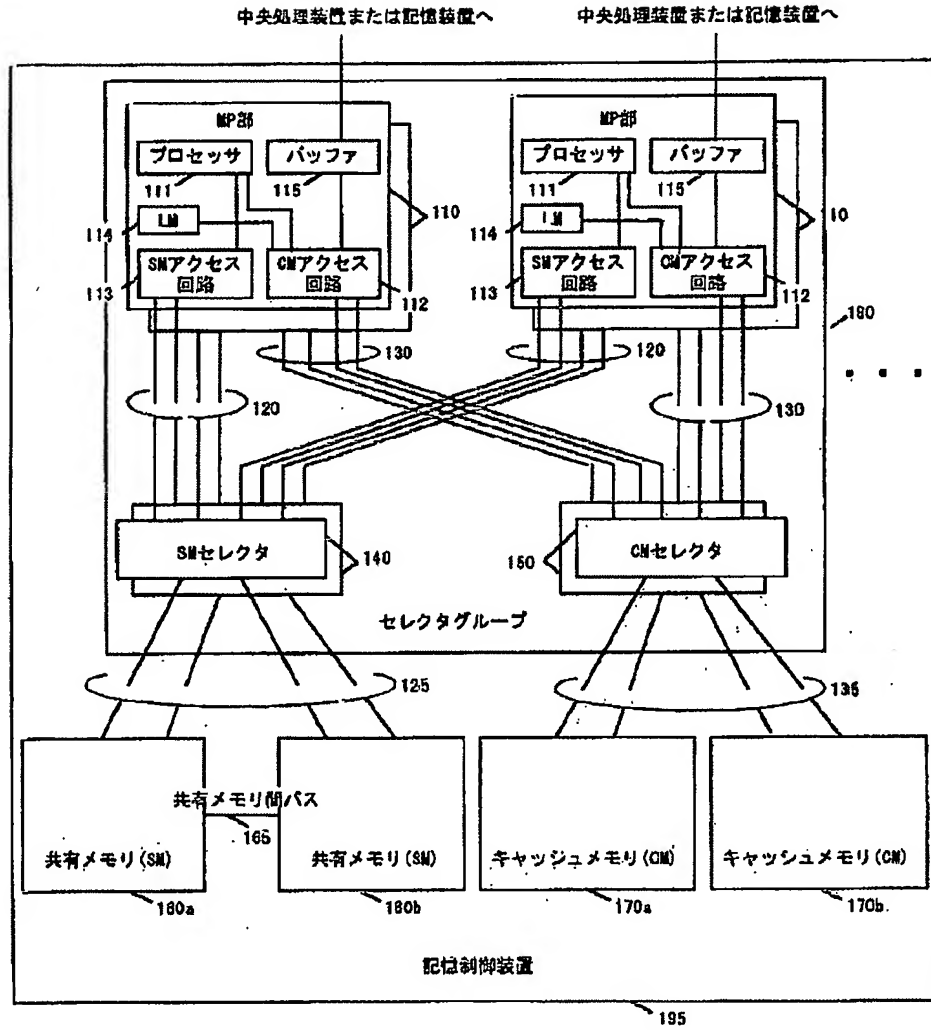
【図2】

図2



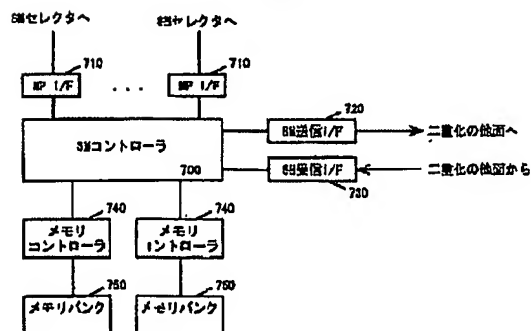
【図1】

図1



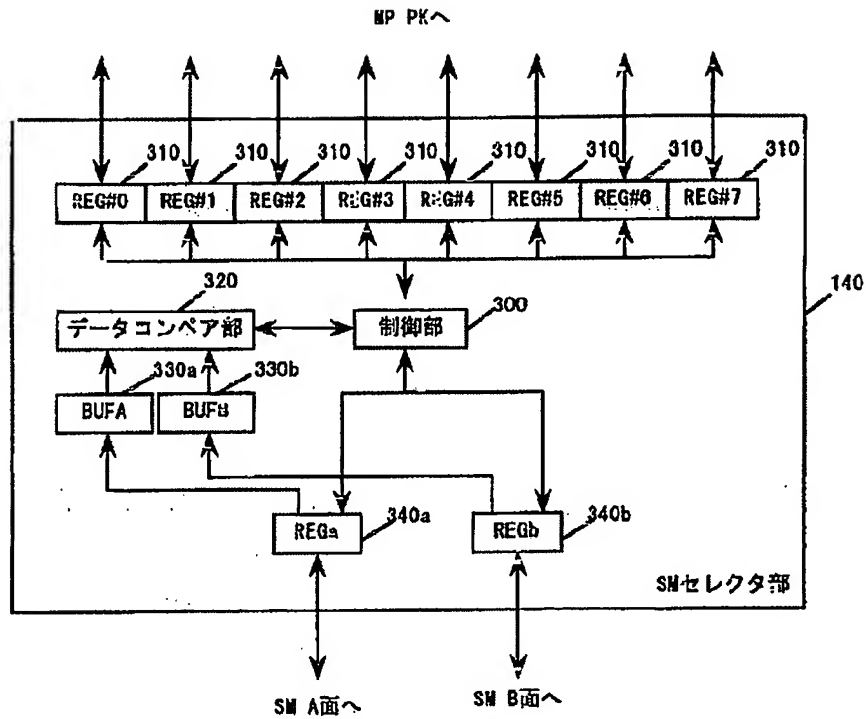
【図7】

図7



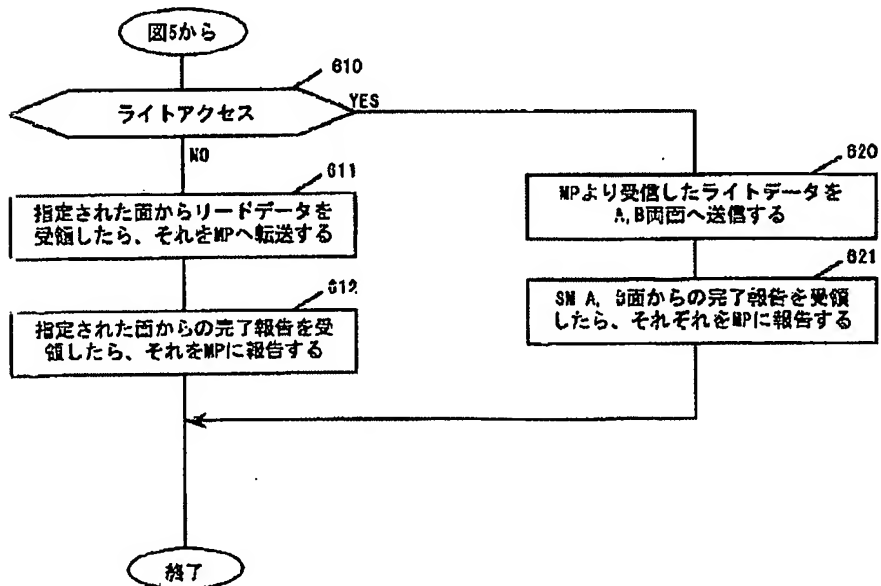
【図3】

図3



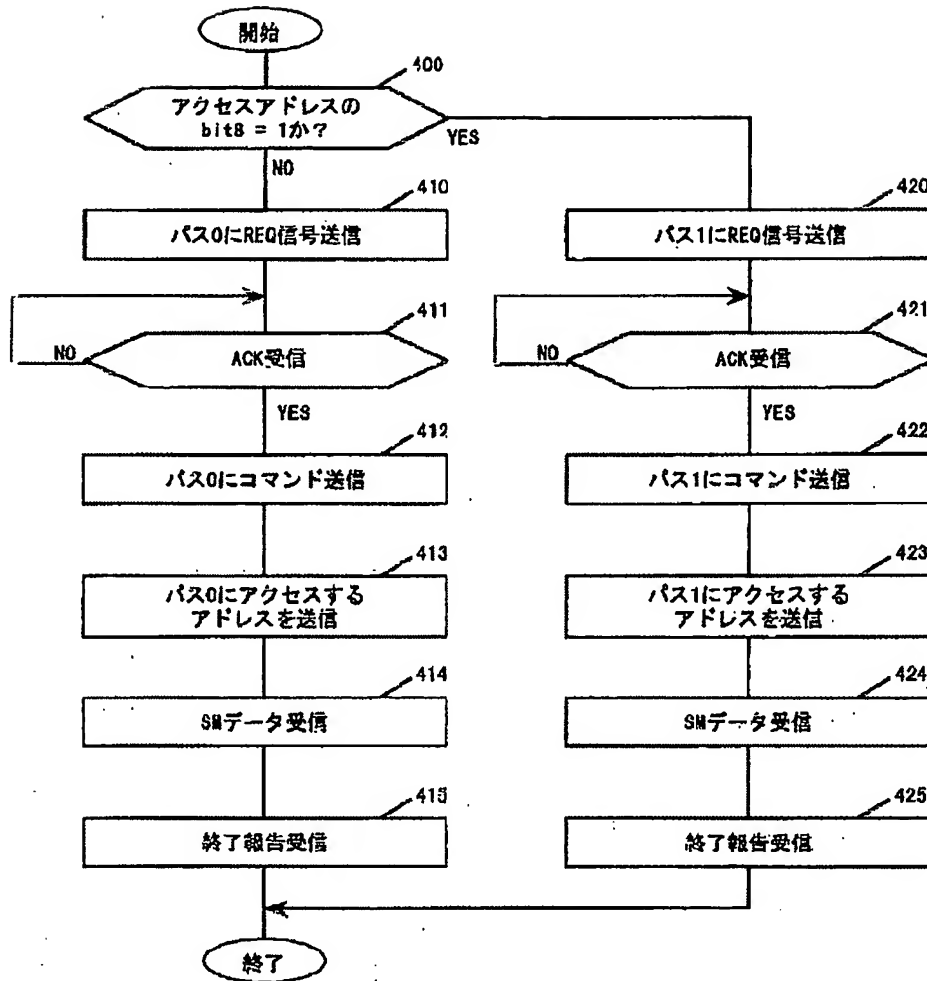
【図6】

図6



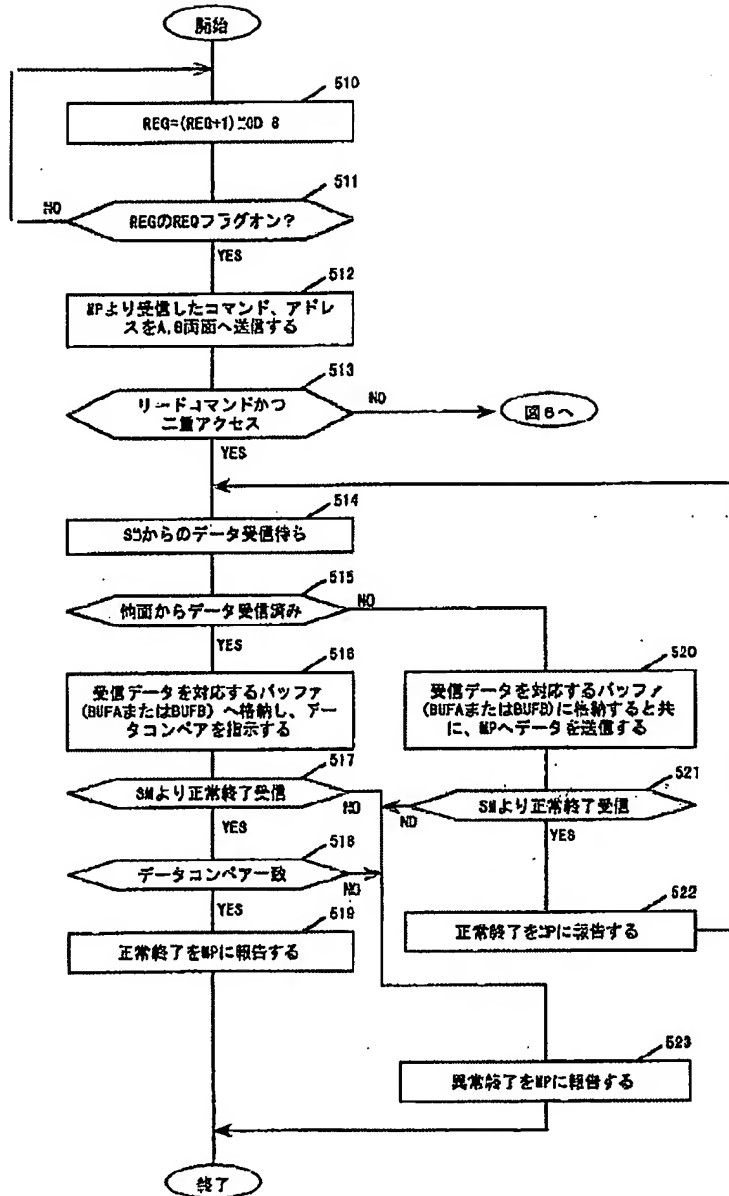
【図4】

図4



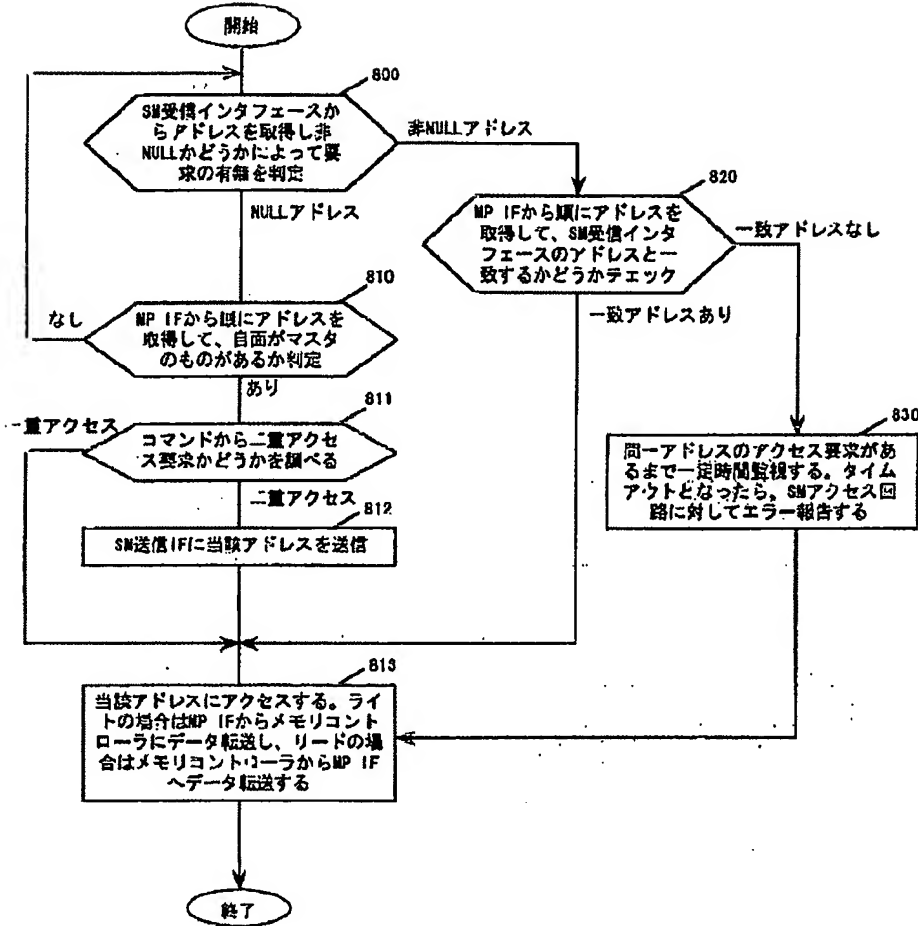
【図5】

図5



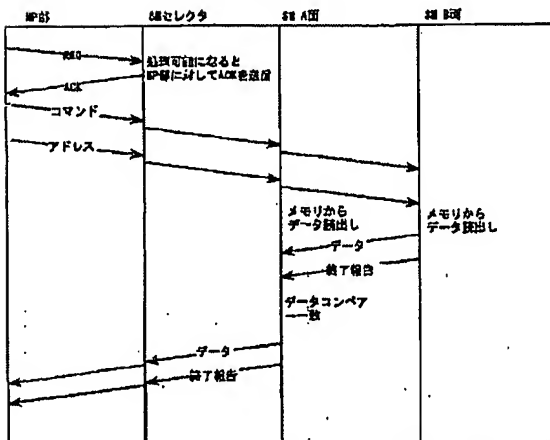
【図8】

図8



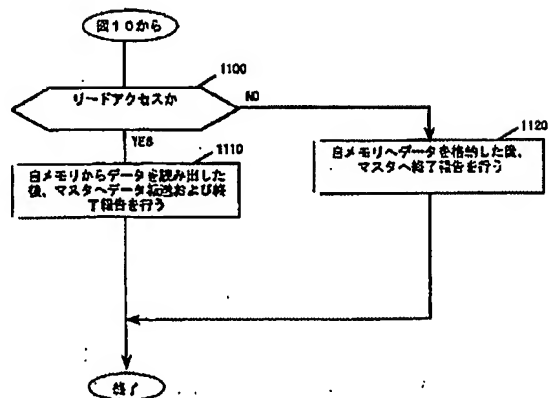
【図9】

図9



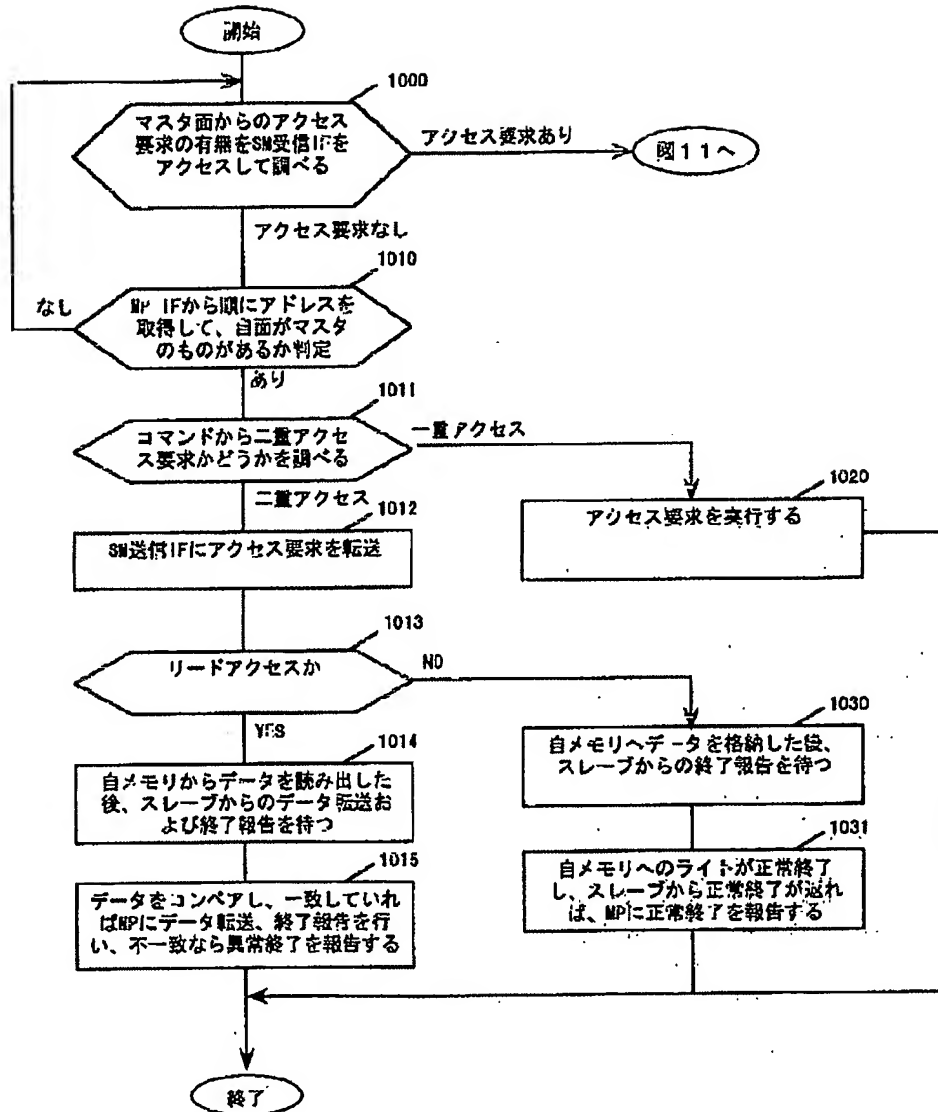
【図11】

図11



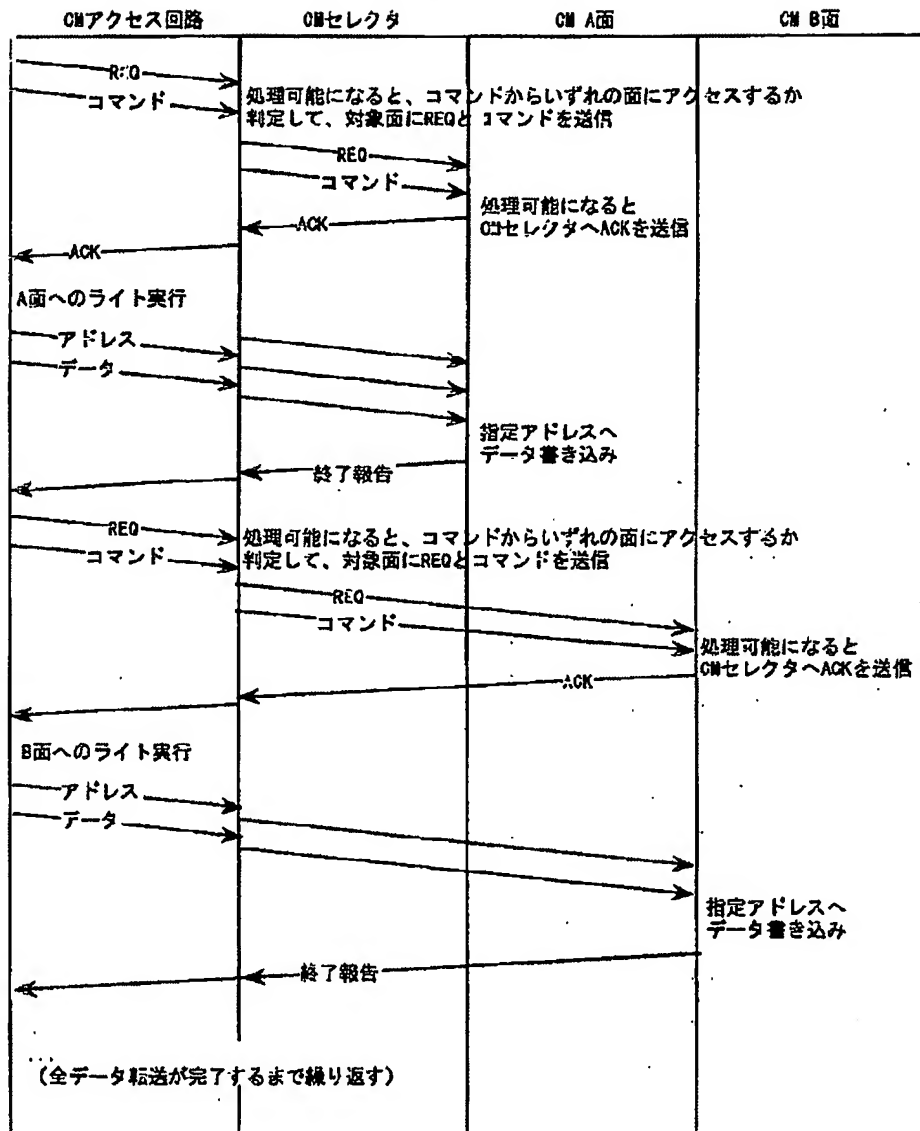
【図10】

図10



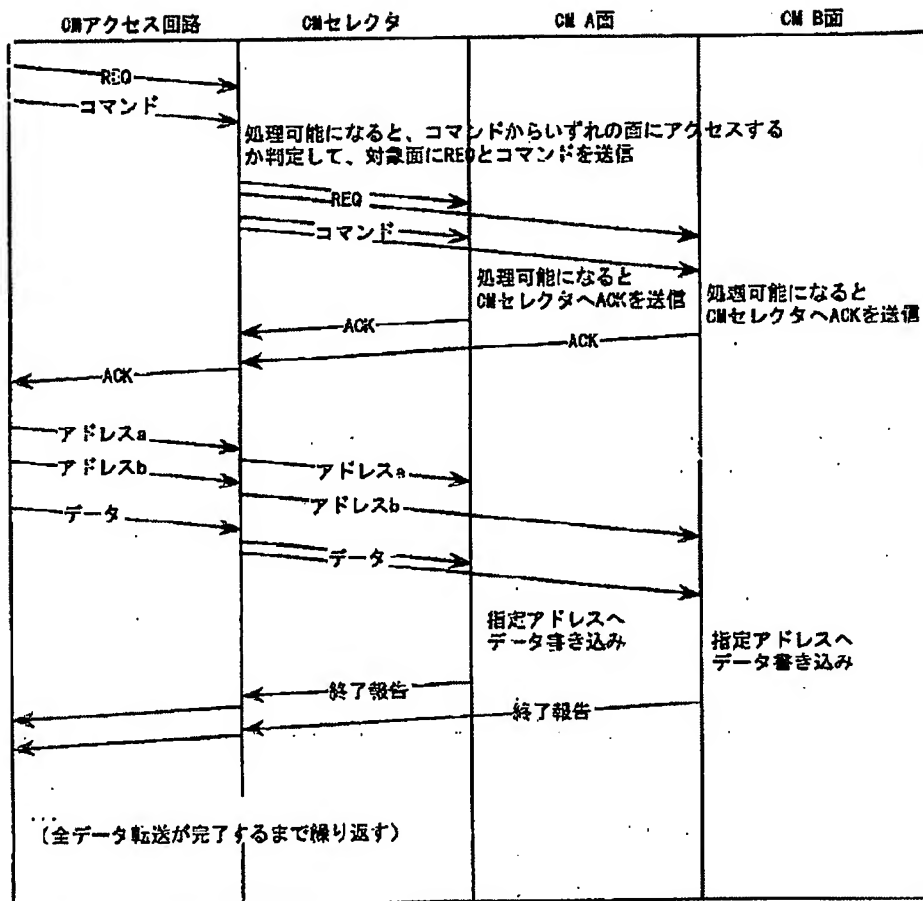
【図12】

図12



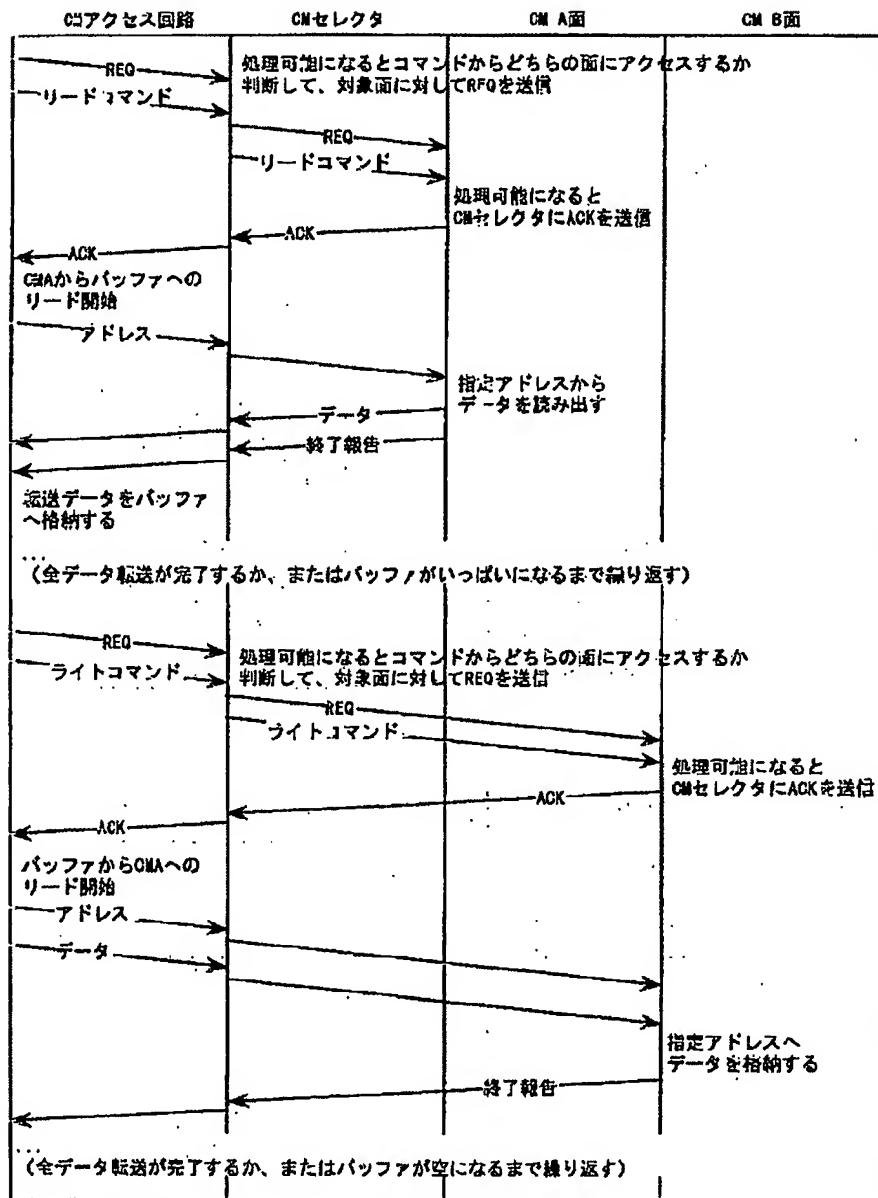
【図13】

図13



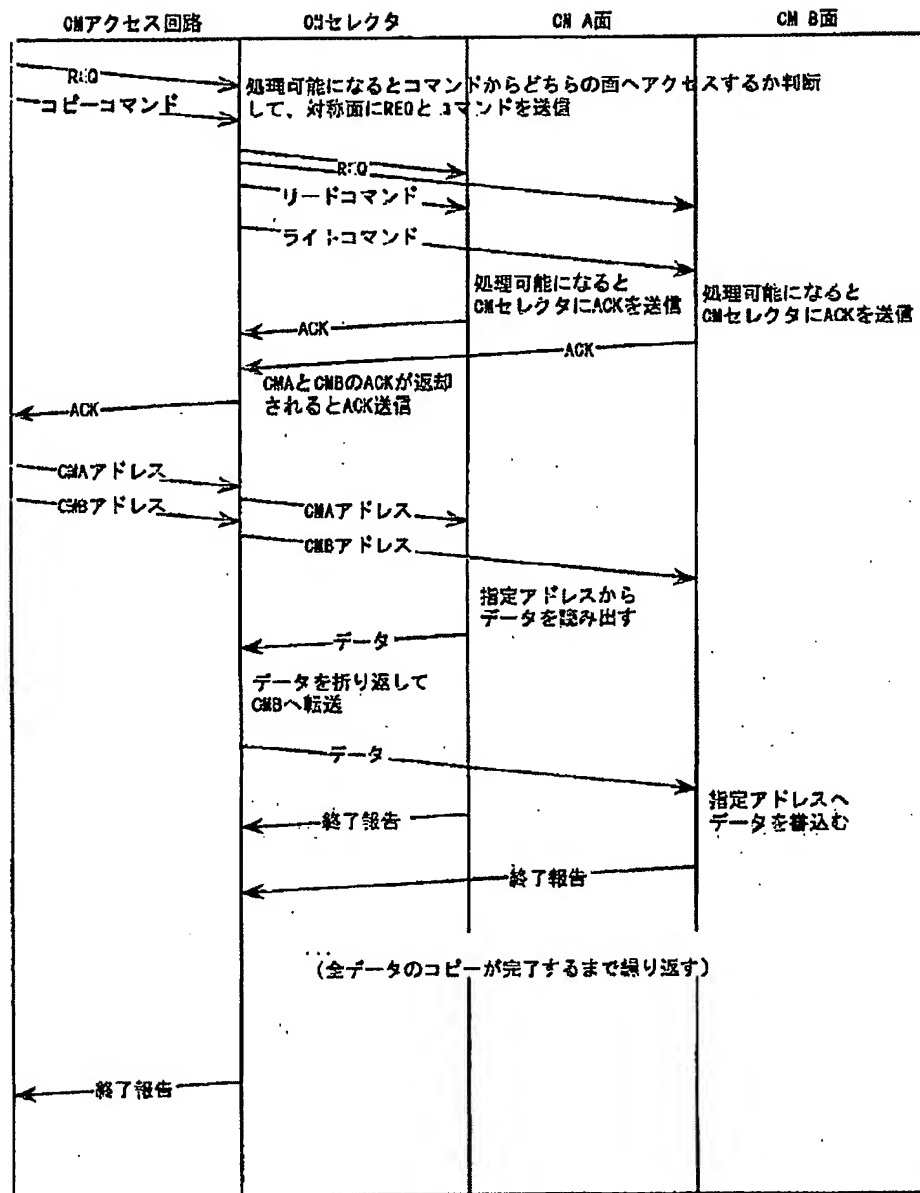
【図14】

図14



【図15】

図15



フロントページの続き

(72)発明者 本間 久雄
 神奈川県小田原市国府津2880番地 株式会
 社日立製作所ストレージシステム事業部内